AD A121845

① 〰

DOCUMENTATION OF
DECISION-AIDING SOFTWARE:

OPINT SYSTEM SPECIFICATION

DECISIONS AND DESIGNS INC.

Dorothy M. Amey
Phillip H. Feuerwerger
Roy M. Gulick

April 1979

N00014-79-C-0069

DTIC

NOV 2 9 1982

S   D

H

ADVANCED ⊕ARPA
DECISION TECHNOLOGY
PROGRAM

CYBERNETICS TECHNOLOGY OFFICE
DEFENSE ADVANCED RESEARCH PROJECTS AGENCY
Office of Naval Research • Engineering Psychology Programs
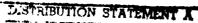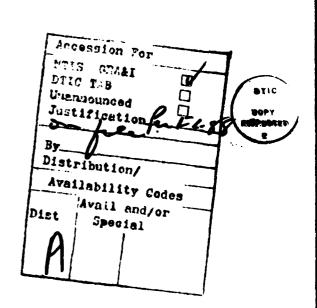
82 11 26 190

DOCUMENTATION OF DECISION-AIDING SOFTWARE:
OPINT SYSTEM SPECIFICATION

by

Dorothy M. Amey, Phillip H. Feuerwerger, and Roy M. Gulick

April 1979

DTIC

NOV 2 9 1982

H

DECISIONS and DESIGNS, INC.

CONTENTS

FIGURES

iii

1.0 INTRODUCTION

1.1 Purpose of the System Specification

The OPINT System Specification is a technical document written for software development personnel. Together with the OPINT Functional Description, it guides the software development effort by identifying the functional requirements and by providing structured logic diagrams that depict the flow, control, and processing of information within the system.

The system specification is generic and is intended to guide and facilitate the preparation of the language-specific program documentation and coding that are necessary to implement and operate OPINT at an installation.

1.2 References

1.2.1 IBM, HIPO--A Design Aid and Documentation Technique. Technical Publication GC20-1851-0. White Plains, New York: IBM, October 1974.

1.2.2 Amey, Dorothy M.; Feuerwerger, Phillip H.; Gulick, Roy M. Documentation of Decision-Aiding Software: OPINT Functional Description. McLean, Virginia: Decisions and Designs, Inc., April 1979.

1.2.3 Amey, Dorothy M.; Feuerwerger, Phillip H.; Gulick, Roy M. Documentation of Decision-Aiding Software: OPINT Users Manual. McLean, Virginia: Decisions and Designs, Inc., April 1979.

1.3 Terms

1.3.1 OPINT - OPINT is an abbreviation for Operations and Intelligence, reflecting the system's major area of applicability.

1.3.2 HIPO - The specification uses the standard Hierarchy plus Input-Process-Output (HIPO) diagramming technique to depict the structural design and logical flow of the system. A legend explaining the HIPO diagramming symbols is included. Reference 1.2.1 provides a complete description of the HIPO documentation technique.

2.0 DESIGN DETAILS

2.1 Background

Systems development personnel should refer to the OPINT
Functional Description, Reference 1.2.2, in conjunction
with the documentation contained in this specification. The
functional description details the decision model implemented
by OPINT and discusses the specific functions that the soft-
ware performs. In addition, systems development personnel
may wish to refer to the OPINT User's Manual, Reference 1.2.3.

2.2 General Operating Procedures

OPINT is a menu-driven system. That is, the system is
designed to interact with the user by presenting a sequential
hierarchy of menus and asking the user to respond by selecting
one option from the current menu. If the user does not select
one of the menu options, the system displays the previous
menu. In this manner, the user moves up and down the hier-
archy, as desired. Whenever data entry is required as a
result of option selection, the system specifically requests
the data and specifies the format.

The system is also designed to be generally forgiving
of procedural errors by the user.

2.3 System Logical Flow

OPINT is a hierarchically structured, modular system.
The system structure and logical flow lends itself to pre-
sentation in the form of HIPO diagrams, which are contained
in this document.

The main purpose of the HIPO diagrams is to provide, in a pictorial manner, the complete set of modular elements necessary to the operation of OPINT including all input, output, and internal functional processing. This is done by displaying input items to the process step which uses them, defining the process, and showing the resulting output of the process step.

The documentation diagrams are designed and drawn in a hierarchical fashion from the main calling routines to the detail-level operation/calculation routines. Extended written descriptions are given below a HIPO diagram whenever it is deemed necessary.

A complete explanation of the symbolic notation used in the HIPO diagrams is given in Reference 1.2.1. An abbreviated legend for the symbols used in this specification is given in Figure 2-1. Note that:

a.   External subroutines appear partly in the process block and partly out. Internal subroutines are shown within the process block.

b.   Overview diagrams show general inputs and outputs only, whereas detail/subroutine-level diagrams show specific input/output tables and/or displays.

c.   Rectangular boxes inside the input/output block areas are generally used to denote single data items. Two or more boxes are grouped to show several data items are input/output.

d.   Rectangular boxes inside the process block indicate repetitive subprocesses.
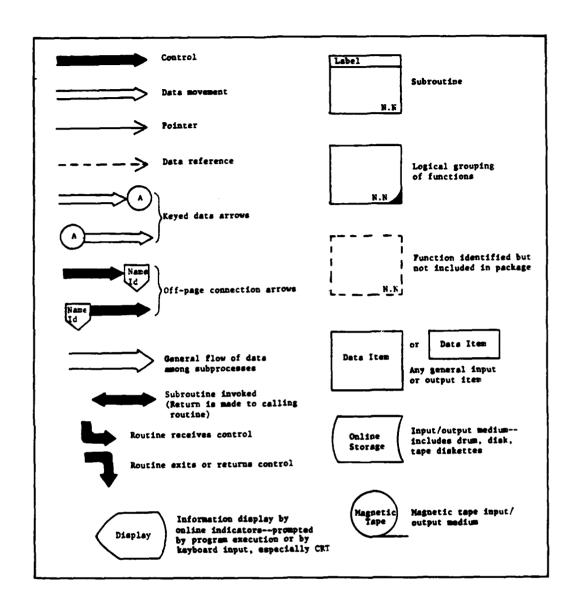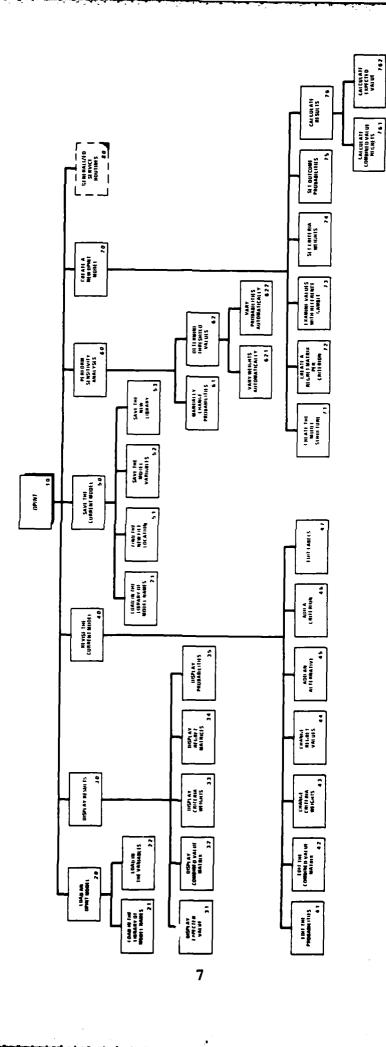
Figure 2-1

LEGEND OF HIPO SYMBOLS

The HIPO diagrams appear in the next section, which completes the system specification.

2.4 HIPO Documentation
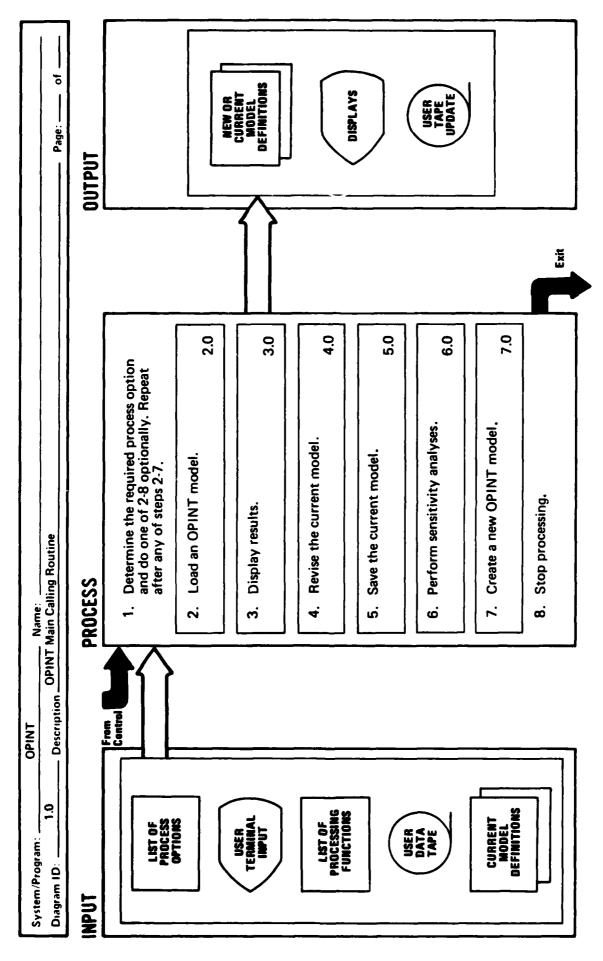
The HIPO diagram identification numbers and figure numbers used in this section stand alone; i.e., they start with 1.0, increase hierarchically, and are independent of the numbering scheme used to this point in this document.

Figure 2-2 is a system structure chart and represents the overall program logic flow in a visual table of contents. The Visual Table of Contents diagram shows the hierarchical structure, the functional description labels, and the diagram (chart) identifiers of functions of OPINT.

Figure 2-2
OPINT OVERVIEW CHART AND VISUAL TABLE OF CONTENTS

PROCESS

1. Determine the required process option and do one of 2-8 optionally. Repeat after any of steps 2-7.

2. Load an OPINT model.   2.0

3. Display results.   3.0

4. Revise the current model.   4.0

5. Save the current model.   5.0

6. Perform sensitivity analyses.   6.0

7. Create a new OPINT model.   7.0

8. Stop processing.

From Control

Exit

OUTPUT

NEW OR CURRENT MODEL DEFINITIONS

DISPLAYS

USER TAPE UPDATE

INPUT

LIST OF PROCESS OPTIONS

USER TERMINAL INPUT

LIST OF PROCESSING FUNCTIONS

USER DATA TAPE

CURRENT MODEL DEFINITIONS

1. In this step, the user is allowed to select from among six different processes. This is done by the display of an options list in menu format and the user's response of inputing a selection via the keyboard display. Once a process is selected, one of steps 2–7 will be invoked to perform the function. If no process is selected (usually indicated by a blank entry from the keyboard), then all processing stops and OPINT returns to the control system.

2-7. These functions are performed only upon selection by the user. See subsequent charts for detail process documentations.
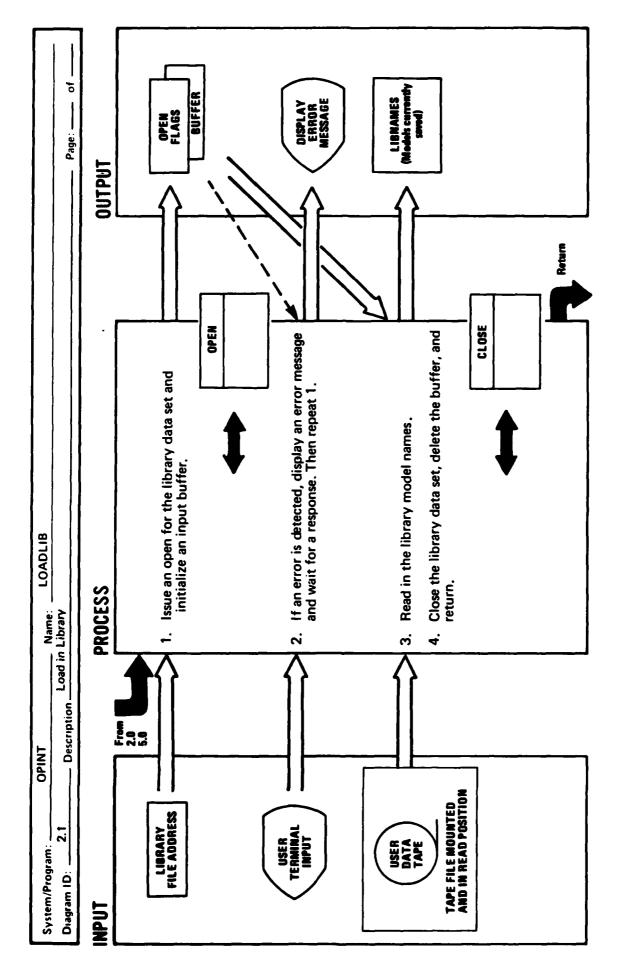
After a process has been selected and the function performed, control returns to step 1. The process options menu is again displayed and a new or the same process may be selected.

INPUT

USER INTERACTION

- USER TAPE FILE — MODEL NAMES
- TERMINAL INPUT
- USER TAPE FILE — MODELS (one per file)

PROCESS

From 1.0

1. Display message to mount the correct tape and wait for a response.

2. Load in the library of model names.

   | LOADLIB | 2.1 |
   | Load in Library | |

   If no models, then do 6.

3. Display the names of the saved models.

4. Determine which file, Y, is to be loaded.

5. Load in the model variables on file Y and do 7.

   | LOADVARS | 2.2 |
   | Load Model | |

6. Display 'No models to load.'

7. Return.

OUTPUT

- DISPLAY 'MOUNT' MESSAGE
- LIBNAMES
- DISPLAY MODEL NAMES
- CURRENT MODEL — MODEL DEFINITIONS
  - Model alternative, outcome, criteria labels, matrices, weights, probabilities
- DISPLAY 'NO MODELS' MESSAGE

Return

---

1. The user may have many tape files on which OPINT formatted models are stored. In this step, the user is prompted for a response indicating the desired tape is mounted or online.
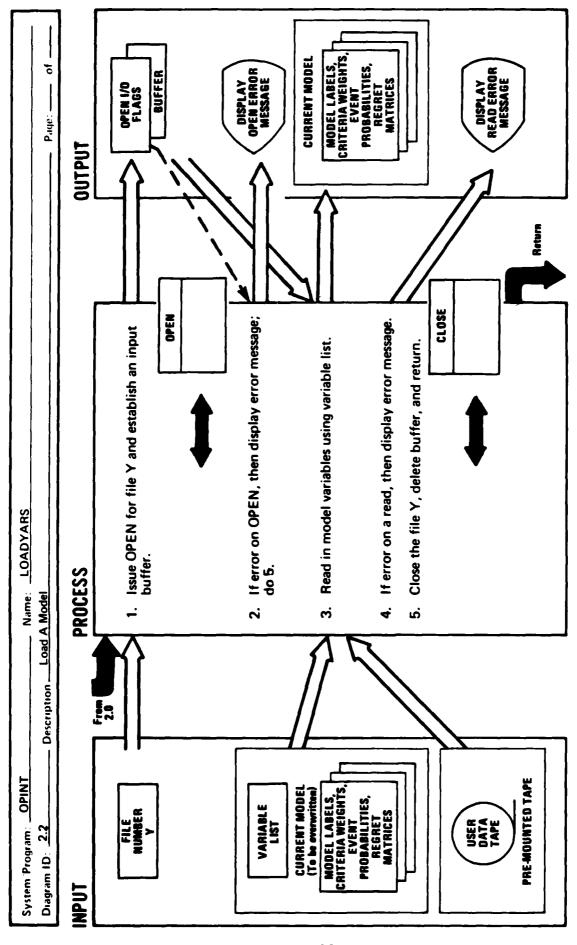
2. The names of the models existing on the mounted tape are displayed in list or MENU format so that the user may select a model for loading.

4. The user is prompted for a model selection: the response may be the list item number or the model name. The requested model is stored in the same tape file as its position relative to the other model names in the displayed list.

9

System/Program: ___OPINT___ Name: ___LOADLIB___

Diagram ID: ___2.1___ Description ___Load in Library___ Page: ___ of ___

INPUT

LIBRARY FILE ADDRESS

USER TERMINAL INPUT

USER DATA TAPE

TAPE FILE MOUNTED AND IN READ POSITION

PROCESS

From 2.0 5.0

1. Issue an open for the library data set and initialize an input buffer.

2. If an error is detected, display an error message and wait for a response. Then repeat 1.

3. Read in the library model names.

4. Close the library data set, delete the buffer, and return.

OPEN

CLOSE

Return

OUTPUT

OPEN FLAGS   BUFFER

DISPLAY ERROR MESSAGE

LIBNAMES (Models currently saved)

1. The library file of model names is available on each OPINT formatted data tape. The file is usually stored and retrieved as a character array and resides on the same device with model data. A system OPEN command is needed to ensure that the data file is online and accessible for reading. An input buffer is needed and provides the link between stored information and program addressable information.

3. The library model names are retrieved from storage. The character array used for holding these model names, LIBNAMES, is of a form which facilitates display; thus, the names may all be of equal character lengths.

4. A system CLOSE command is issued to free the data file for later use.

10

INPUT

FILE NUMBER Y

VARIABLE LIST

CURRENT MODEL
(To be overwritten)

MODEL LABELS, CRITERIA WEIGHTS, EVENT PROBABILITIES, REGRET MATRICES

USER DATA TAPE

PRE-MOUNTED TAPE

PROCESS

From 2.0

1. Issue OPEN for file Y and establish an input buffer.

OPEN

2. If error on OPEN, then display error message; do 5.

3. Read in model variables using variable list.

4. If error on a read, then display error message.

5. Close the file Y, delete buffer, and return.

CLOSE

Return

OUTPUT

OPEN I/O FLAGS

BUFFER

DISPLAY OPEN ERROR MESSAGE

CURRENT MODEL

MODEL LABELS, CRITERIA WEIGHTS, EVENT PROBABILITIES, REGRET MATRICES

DISPLAY READ ERROR MESSAGE

3. A list of variable Names or identifiers is kept so that load and store routines will always process the variables in the same sequence order.

4. The Model variables retrieved from storage are used in all other program functions (see chart 1.0). The variables which must be loaded are the following:

- DECISION ALTERNATIVE LABELS
- EVENT OUTCOME LABELS
- KEY EVENT NAME
- CRITERIA LABELS
- CRITERIA WEIGHT VALUES
- EVENT PROBABILITY VALUES
- REGRET VALUES
- COMPUTED COMBINED-REGRET

11

INPUT

LIST OF OPTIONAL DISPLAYS

LABELS FOR CURRENT MODEL

CURRENT WEIGHTS, PROBABILITIES, MATRICES...

LIST OF DISPLAY FUNCTIONS

USER RESPONSE TERMINAL INPUT

12

From 1.0

PROCESS

1. Display the list of optional displays.

2. Determine item to be displayed. If none, go to 4.

3. Exit to the appropriate subroutine.

   Display Function

4. Return to main program.

Return

OUTPUT

DISPLAY OPTIONAL LISTS

REQUESTED DISPLAY

1. The List of Optional displays is shown in Menu format. The list of displays corresponds exactly with the associated list of display functions.

2. A display selection is requested from the user. If the user responds with a blank entry (no MENU selection), then control returns to the calling routine.

3. The display routine which provides the required display is in the list of display functions which corresponds to the order of optional displays. (See Charts 3.1 – 3.5 for detail descriptions.)

12

INPUT

MODEL NAME, ALTERNATIVE AND OUTCOME LABELS

WEIGHTED EXPECTED-VALUES

USER TERMINAL INPUT

PROCESS

From 3.0

1. Display heading, outcome labels.

2. Display the expected-value matrix and the expected value vector.

SHOWMAT

8.1

3. Display a continue message and wait for a response.

Return

OUTPUT

DISPLAY (See figure 3.1)

DISPLAY SCREEN NO. 1

1. The headings for the display are unique to the model that is currently available. This display should show the model name and labels for outcomes in horizontal fashion.

3. The user is prompted to input any response from the keyboard to cause processing to continue.

13

MODEL – – EVACUATION

EXPECTED VALUE

	NONE	P-300	P-2K	NP-2K	NP-6K	TOTAL
NORMAL	0	0	−4	−13	−10	−28
LOW PROF	−5	0	−2	−12	−10	−29
MED PROF	−12	−1	−3	−5	−5	−26
EVAC PST	−15	−1	−4	−4	−4	−28

Figure 3.1

EXAMPLE OF OUTPUT FROM DISPLAY FUNCTION 3.1

14

System/Program: _____ OPINT _____ Name: _____

Diagram ID: _____ 3.2 _____ Description _____ Display Combined-Value Matrix _____ Page: _____ of _____

INPUT

PROCESS

OUTPUT

From 3.0

MODEL NAME, ALTERNATIVE, OUTCOME LABELS

COMBINED-VALUE REGRET MATRIX OF CURRENT MODEL

USER TERMINAL INPUT

1. Display heading, outcome labels.

2. Display the combined-value matrix.

SHOWMAT

Display Matrix

8.1

3. Display a continue message.

4. Wait for a response.

Return

DISPLAY COMBINED-VALUE REGRETS

DISPLAY SCREEN NO. 1
(See Figure 3.2)

15

MODEL – – EVACUATION

COMBINED VALUE

	NONE	P-300	P-2K	NP-2K	NP-6K
NORMAL	0	−17	−31	−74	−74
LOW PROF	−9	−9	−17	−71	−71
MED PROF	−24	−22	−22	−30	−32
EVAC PST	−30	−26	−26	−26	−26

Figure 3.2

EXAMPLE OF OUTPUT FROM DISPLAY FUNCTION 3.2

INPUT

MODEL LABELS

CURRENT WEIGHTS

USER TERMINAL INPUT

PROCESS

From 3.0

1. Display heading.

2. Display the criteria names.

3. Display the weights under respective criteria labels.

4. Display a continue message.

5. Wait for a response.

Return

OUTPUT

DISPLAY

DISPLAY SCREEN NO. 1

1. The display heading should include the Model name.

17

INPUT

- CRITERIA LABELS
- INTERACTIVE MODE
  - USER TERMINAL INPUT
- CURRENT MODEL
  - CRITERIA, OUTCOME, AND OPTION LABELS, WEIGHTS
- REGRET MATRICES
- USER TERMINAL INPUT

PROCESS

From 3.0

1. Display list of regret matrices.

2. Determine which matrix is to be displayed.

   If none, return.

3. Display criterion name and weight.

4. Display the matrix requested.

   SHOWMAT
   Display a Matrix

5. Display a continue message and wait for a response.

   Repeat from step 1.

OUTPUT

- DISPLAY OPTIONAL LIST
- DISPLAY SCREEN NO. 1
- CRITERION NAME FOR DISPLAY
- Return
- DISPLAY HEADING, WEIGHT, MATRIX
- DISPLAY
- DISPLAY SCREEN NO. 2

1. The list of Regret Matrix names is shown in MENU format.

2. The user inputs his selection for display. A blank input entry causes a return to the calling routine.

3.4. The requested matrix is displayed along with the appropriate alternative, outcome, and criterion labels. (SHOWMAT, Chart 8.1)

5. A message is written underneath the matrix display which tells the user that a keyboard response will cause the current screen display to vanish. (WAIT, Chart 8.2)
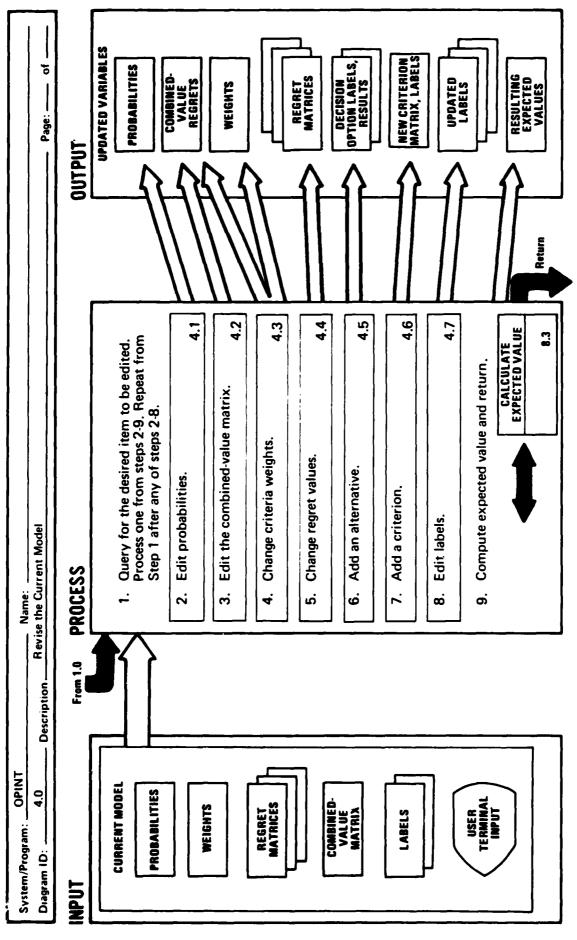
The processing can be repeated for other matrices by the return to step 1.

INPUT

MODEL NAME, OUTCOME LABELS

CURRENT PROBABILITIES

USER TERMINAL INPUTS

PROCESS

From 3.0

1. Display heading for the current model.

2. Display the names of event outcomes.

3. Display the probabilities under the respective outcome names.

4. Display a continue message.

5. Wait for a response.

OUTPUT

DISPLAY HEADINGS, PROBABILITIES OF OUTCOMES

DISPLAY

DISPLAY SCREEN NO. 1

Return

The display should show the model name as a heading, the unique outcome labels horizontally, and the respective outcome probability values below the outcome labels.

INPUT

CURRENT MODEL

- PROBABILITIES
- WEIGHTS
- REGRET MATRICES
- COMBINED-VALUE MATRIX
- LABELS
- USER TERMINAL INPUT

PROCESS

From 1.0

1. Query for the desired item to be edited. Process one from steps 2-9. Repeat from Step 1 after any of steps 2-8.

2. Edit probabilities. — 4.1

3. Edit the combined-value matrix. — 4.2

4. Change criteria weights. — 4.3

5. Change regret values. — 4.4

6. Add an alternative. — 4.5

7. Add a criterion. — 4.6

8. Edit labels. — 4.7

9. Compute expected value and return.

CALCULATE EXPECTED VALUE   8.3

Return

OUTPUT

UPDATED VARIABLES

- PROBABILITIES
- COMBINED-VALUE REGRETS
- WEIGHTS
- REGRET MATRICES
- DECISION OPTION LABELS, RESULTS
- NEW CRITERION MATRIX, LABELS
- UPDATED LABELS
- RESULTING EXPECTED VALUES

1. The list of model definition items available for editing or revision is displayed in Menu format. The user is then prompted for a selection which causes one of steps 2 through 9 to be executed.

2-8. These subprocesses are explained in the next seven charts as indicated in the diagram (4.1 – 4.7).

9. Always, the expected values are recomputed as the final result after editing.

The output from the edit processes effect a permanent change in the model variables which are updated as long as the program OPINT is in operation. To secure these changes, the user must store the new values via the SAVE A MODEL processing option (See chart 5.0).

20

INPUT

CURRENT PROBS

OUTCOME LABELS

USER TERMINAL INPUT

PROCESS

From 4.0

1. Display the current probabilities.

2. Prompt the user for updates and read in updates.

3. Normalize the updated probabilities and display.

4. If corrections desired, repeat from 1.

5. Permanently change the current probabilities.

Return

OUTPUT

DISPLAY PROBS

PROB. UPDATE

NORMALIZED UPDATED PROBS

DISPLAY UPDATE

CURRENT PROBS

1. Display the current outcome probability values with/below the respective unique outcome labels.

3. The updates which are input by the user are normalized (divided by the sum of the updates and then multiplied by 100) to sum to 100. The resulting numbers are displayed below the input values.

4. The user is queried for corrections. At this point, the current probabilities have not been disturbed.
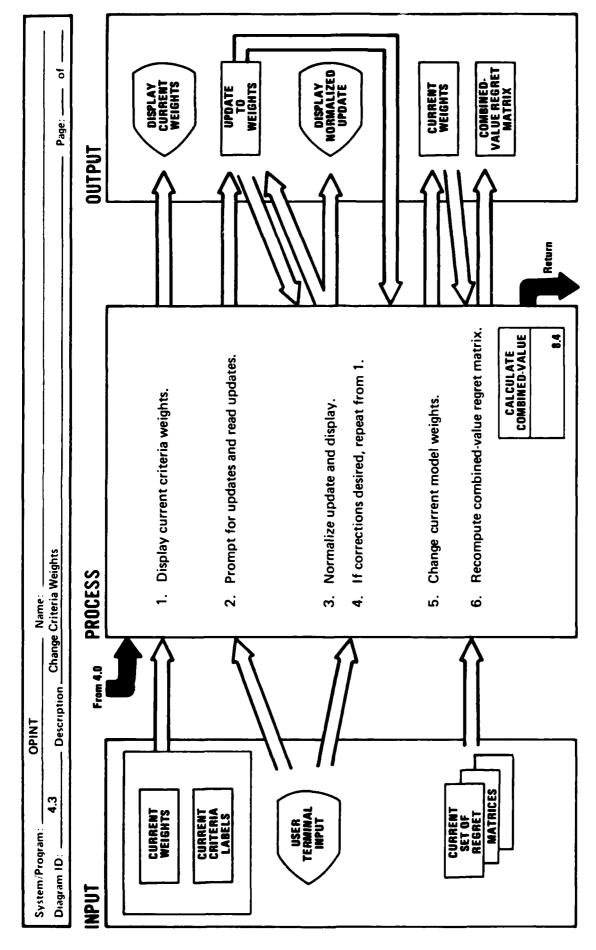
5. After all corrections are made to the updated values, the current set of outcome probabilities is permanently updated.

INPUT

CURRENT COMBINED-VALUE MATRIX

MODEL LABELS

USER TERMINAL INPUT

From 4.0

PROCESS

1. Display the current combined value regret matrix.

2. Determine which row/column is to be updated. If none, do step 7.

3. Display the row or column.

4. Prompt for update.

5. Read in update. If user wants to correct, repeat from 3.

6. Change row or column of matrix.

7. Return.

OUTPUT

DISPLAY MATRIX (Figure 3.2)

ROW/COLUMN NO., LABEL

DISPLAY FOR UPDATE

UPDATED ROW/COLUMN

(Updated) COMBINED-VALUE REGRET MATRIX

1. Display the combined-value regret matrix with outcome and alternative labels (see chart 3.2 and figure 3.2).

2. This step begins an iterative process. The user is prompted for the name of the row (alternative) or column (outcome) to be updated. If no name is specified, then the process ends. Otherwise, steps 3-6 are executed.
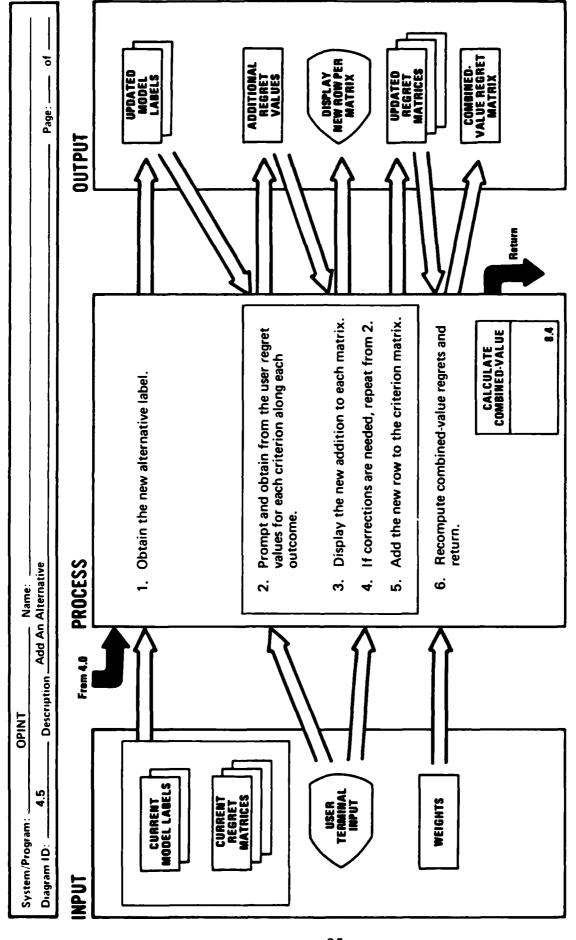
6. The input update values are used to permanently change (until later updates to regrets or criteria weights) the combined-value regrets.

INPUT

- CURRENT WEIGHTS
- CURRENT CRITERIA LABELS
- USER TERMINAL INPUT
- CURRENT SET OF REGRET MATRICES

PROCESS

From 4.0

1. Display current criteria weights.

2. Prompt for updates and read updates.

3. Normalize update and display.

4. If corrections desired, repeat from 1.

5. Change current model weights.

6. Recompute combined-value regret matrix.

CALCULATE COMBINED-VALUE
8.4

OUTPUT

- DISPLAY CURRENT WEIGHTS
- UPDATE TO WEIGHTS
- DISPLAY NORMALIZED UPDATE
- CURRENT WEIGHTS
- COMBINED-VALUE REGRET MATRIX

Return

5. At this step, final corrections to the updates are used to permanently change the set of criteria weights.

INPUT

- USER TERMINAL INPUT
- MODEL LABELS
- REGRET MATRICES
- USER TERMINAL INPUT
- WEIGHTS

PROCESS

1. Determine which criterion regret matrix is to be updated. If none, do 8.

2. Display the regret matrix.

3. Determine the row or column to be updated; if none, do 1.

4. Display the row/column.

5. Prompt, read update to row/column.

   If corrections needed, repeat from 4.

6. Change current matrix row/column.

7. Recompute combined-value regret matrix.

8. Return.

OUTPUT

- CRITERION NAME
- DISPLAY MATRIX
- ROW/COL NAME
- DISPLAY ROW OR COLUMN
- UPDATE ROW/COLUMN
- UPDATED MATRIX
- COMBINED-VALUE REGRETS

Return

From 4.0

1. Prompt the user for the name of criterion matrix to be updated. If no name is specified, then return to the calling routine. Otherwise continue processing.

2. Display regret values along with the appropriate alternative and outcome labels.

3. In repetitive manner, prompt the user for the row or column name (or number) to be modified. If no row or column is given, then exit from the inner loop and repeat step 1. Otherwise, execute steps 3-6.

6. Do not permanently change the selected regret matrix row/column until this step in the procedure.

7. Subroutine invoked for this step is documented in chart 8.4.

24

INPUT

PROCESS

OUTPUT

From 4.0

INPUT boxes:
- CURRENT MODEL LABELS
- CURRENT REGRET MATRICES
- USER TERMINAL INPUT
- WEIGHTS

PROCESS:

1. Obtain the new alternative label.

2. Prompt and obtain from the user regret values for each criterion along each outcome.

3. Display the new addition to each matrix.

4. If corrections are needed, repeat from 2.

5. Add the new row to the criterion matrix.

6. Recompute combined-value regrets and return.

CALCULATE COMBINED-VALUE

8.4

Return

OUTPUT boxes:
- UPDATED MODEL LABELS
- ADDITIONAL REGRET VALUES
- DISPLAY NEW ROW PER MATRIX
- UPDATED REGRET MATRICES
- COMBINED-VALUE REGRET MATRIX

1. Prompt the user for the new alternative label and update the list of alternative labels with the new label.
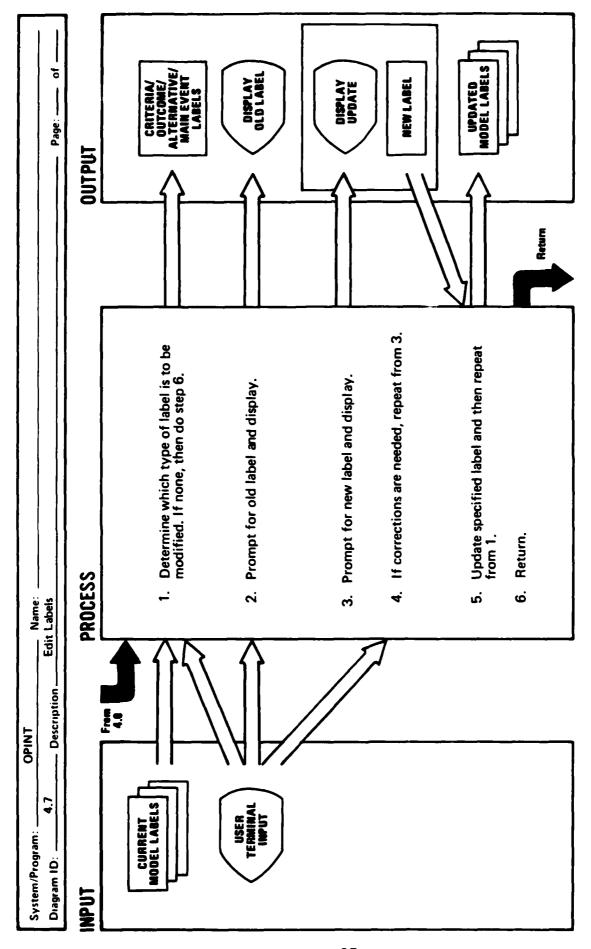
2. For each criterion, prompt the user for numerical regret value input. This is done by displaying the event outcome labels horizontally to remind the user of the Matrix cell position and prompting values along the outcomes for the new alternative.

25

INPUT

USER INPUT

USER INPUT

CURRENT REGRET MATRICES

PROCESS

1. Obtain the new criterion label.

2. Obtain a complete set of regrets for this criterion.

3. Display the new matrix.

4. If corrections are needed, repeat from 2.

5. Obtain a new set of weights.

| SET CRITERIA |
| Weights | 7.4 |

6. Recompute combined-value regrets and return.

From 4.0

OUTPUT

NEW CRITERIA LABELS

NEW REGRET MATRIX

DISPLAY MATRIX

WEIGHTS

COMBINED-VALUE MATRIX

16

16

Return

1. The user is prompted for a new criterion name to be input from the keyboard display. The criterion name is added to the list of criteria labels.
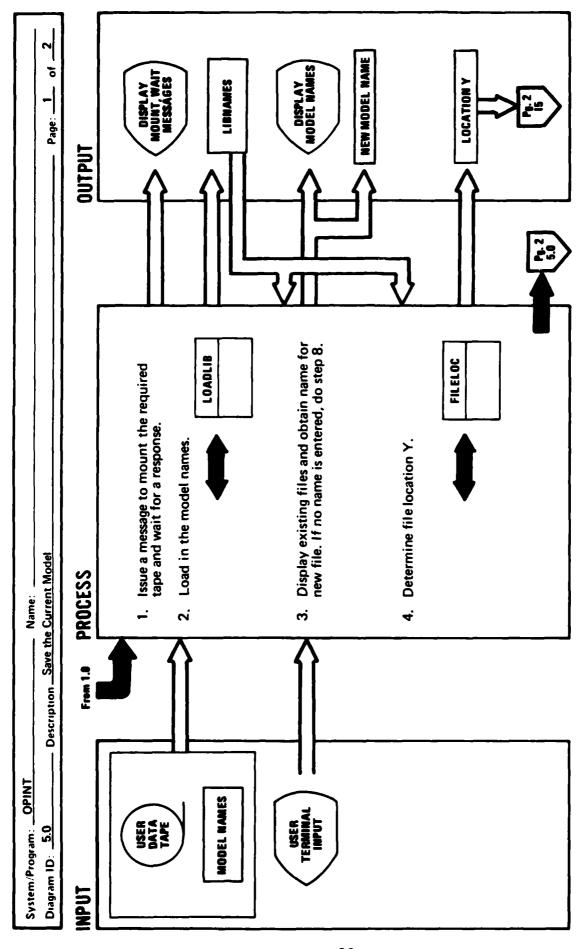
2. For each alternative, prompt the user for a regret value along each event outcome. The event outcome labels should appear on the display as a heading.

5. A new set of weights are input via the keyboard for all criteria in the model. (see chart 7.4)

6. The subroutine invoked for this step is documented in chart 8.4.

INPUT

CURRENT MODEL LABELS

USER TERMINAL INPUT

PROCESS

From 4.0

1. Determine which type of label is to be modified. If none, then do step 6.

2. Prompt for old label and display.

3. Prompt for new label and display.

4. If corrections are needed, repeat from 3.

5. Update specified label and then repeat from 1.

6. Return.

OUTPUT

CRITERIA/ OUTCOME/ ALTERNATIVE/ MAIN EVENT LABELS

DISPLAY OLD LABEL

DISPLAY UPDATE

NEW LABEL

UPDATED MODEL LABELS

Return

1. Display the label types which can be modified in a Menu list. Prompt the user for the type desired. If no valid response, then return to the calling routine. Otherwise do steps 2-5.

5. At this step labels are permanently modified for the operating program.

27

INPUT

USER DATA TAPE

MODEL NAMES

USER TERMINAL INPUT

PROCESS

From 1.0

1. Issue a message to mount the required tape and wait for a response.

2. Load in the model names.

LOADLIB

3. Display existing files and obtain name for new file. If no name is entered, do step 8.

4. Determine file location Y.

FILELOC

OUTPUT

DISPLAY MOUNT WAIT MESSAGES

LIBNAMES

DISPLAY MODEL NAMES

NEW MODEL NAME

LOCATION Y

Pg. 2 I5

Pg. 2 5.0

1. The computer program prompts for an indication that the desired storage file/device has been selected and placed online. Any response from the keyboard causes processing to resume.
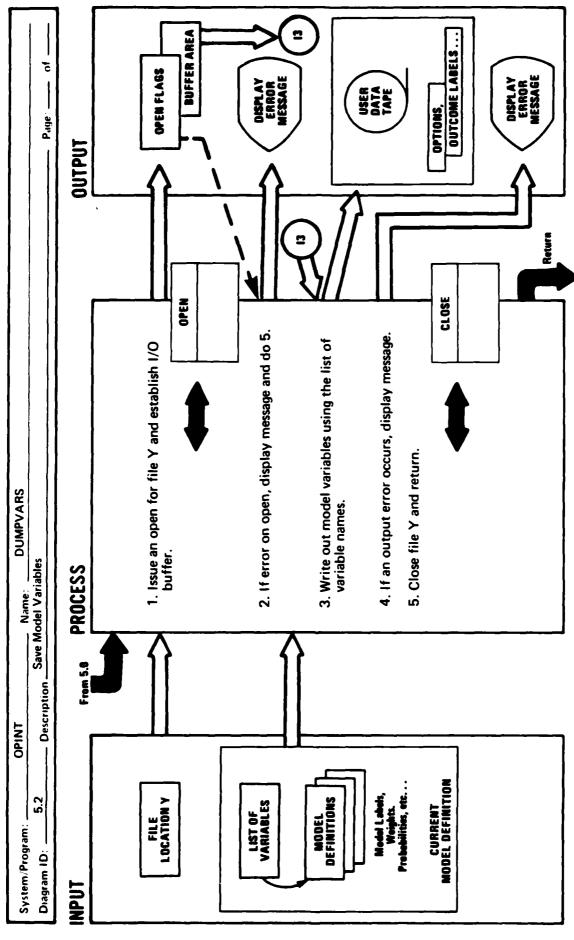
4. The existing file structure and the amount of available space on the data tape are checked along with the user specification to determine where the model variables are to be stored.

28

INPUT

VARIABLE LIST

MODEL DEFINITIONS
Model Labels, Weights, Probabilities, etc. . .

LIBNAMES

PROCESS

From Pg. 1 5.0

5. If location Y = 0, then repeat 2. Otherwise, save model on file Y.

DUMPVARS

6. Add or replace new file name in list of saved models.

7. Save the new list of existing models.

SAVELIB

8. Return.

OUTPUT

Pg. 1 I5

USER DATA TAPE

MODEL VARIABLES (one per record)

NEW MODEL NAME

UPDATED LIBNAMES

USER DATA TAPE

PRE-MOUNTED AND POSITIONED TO FILE ADDRESS

Return

6. The library name list is updated to reflect the new file structure. The new model name's position in the LIBNAMES array must be the same relative position to other models stored on the device.

INPUT

LIBNAMES

NEW NAME

USER TERMINAL INPUT

PROCESS

From 5.0

1. If new name already exists in the library, then display a message.

   a. If user wants to replace the file, set Y = current file location. Do step 4.

   b. If no replacement wanted, set Y = 0 and do 4.

2. If there are no existing models, set Y = 1 and do step 4.

3. Set Y to one plus the number of models currently saved.

4. Return.

OUTPUT

DISPLAY 'DUPLICATE NAME' MESSAGE

FILE LOCATION
Y = CURRENT LOCATION OR Y = 0

FILE LOCATION Y = 1

FILE LOCATION Y = NO. MODELS +1

Return

INPUT

FILE LOCATION Y

LIST OF VARIABLES

MODEL DEFINITIONS

Model Labels, Weights, Probabilities, etc....

CURRENT MODEL DEFINITION

PROCESS

From 5.0

1. Issue an open for file Y and establish I/O buffer.

OPEN

2. If error on open, display message and do 5.

3. Write out model variables using the list of variable names.

4. If an output error occurs, display message.

5. Close file Y and return.

CLOSE

Return

OUTPUT

OPEN FLAGS

BUFFER AREA

I3

DISPLAY ERROR MESSAGE

USER DATA TAPE

OPTIONS, OUTCOME LABELS...

DISPLAY ERROR MESSAGE

I3

1. The file location Y is used to determine an exact storage position on the selected device.
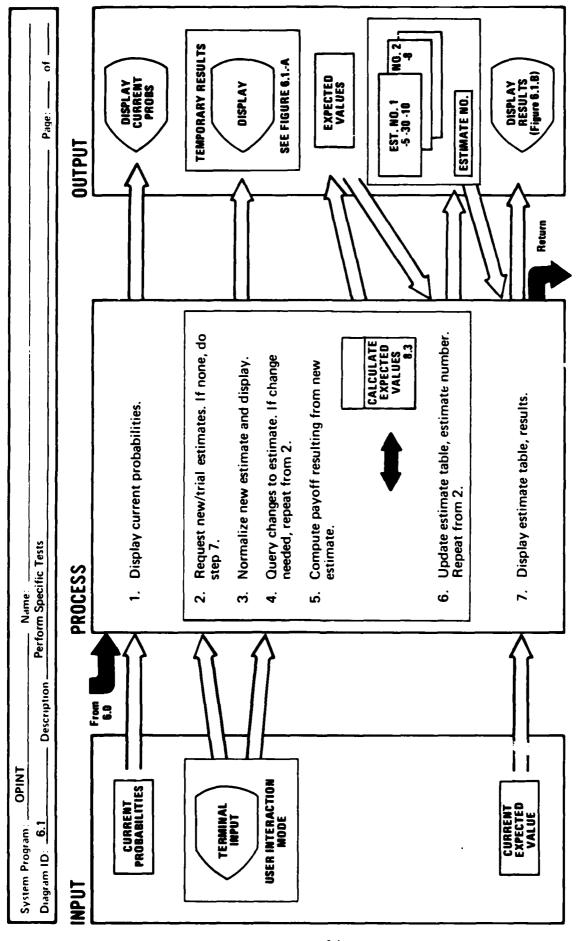
3. The list of variable names is identical to the list of names used to Load a Model. (see Chart 2.2)

31

INPUT

LIBRARY FILE ADDRESS

USER TERMINAL INPUT

USER DATA TAPE

FILE MOUNTED AND IN WRITE POSITION

PROCESS

From 5.0

1. Issue an open for output to the library data set.

2. If an error is detected, display an error message. Wait for a response and then repeat 1.

3. Write out library file names.

4. Close the library data set.

5. If error is detected, display message, wait for a response.

OPEN

CLOSE

Return

OUTPUT

I/O FLAGS, BUFFER

USER DATA TAPE

MODEL NAMES

I/O FLAGS

32

INPUT

- USER TERMINAL INPUT
- PROBABILITIES
- WEIGHTS
- CURRENT EXPECTED-VALUE AND COMBINED-VALUE MATRICES

PROCESS

From 1.0

1. Determine the type of analysis to be performed. If none, do step 4; otherwise do 2 or 3 and repeat from 1.

2. Manually change probabilities to specific values.   6.1

3. Automatically determine threshold values.   6.2

4. Return.

Return

OUTPUT

- DISPLAY OPTIONS MENU
- DISPLAY NEW RESULTS (See Figure 6.1-B)
- DISPLAY RESULTS, THRESHOLD ARROWS (See Figure 6.2.2)

1. Display the types of processing available via a Menu list and prompt the user for his choice of operation. If no analysis is desired, return to the calling routine. Otherwise, perform step 2 (chart 6.1) or step 3 (chart 6.2) and repeat the display and selection prompting.

All changes made in steps 2 and 3 are temporary and they are used only for display purposes.

33

System Program: __OPINT__   Name: _____

Diagram ID: __6.1__   Description __Perform Specific Tests__   Page: ___ of ___

INPUT

- CURRENT PROBABILITIES
- TERMINAL INPUT
- USER INTERACTION MODE
- CURRENT EXPECTED VALUE

From 6.0

PROCESS

1. Display current probabilities.

2. Request new/trial estimates. If none, do step 7.

3. Normalize new estimate and display.

4. Query changes to estimate. If change needed, repeat from 2.

5. Compute payoff resulting from new estimate.

CALCULATE EXPECTED VALUES 8.3

6. Update estimate table, estimate number. Repeat from 2.

7. Display estimate table, results.

OUTPUT

- DISPLAY CURRENT PROBS
- TEMPORARY RESULTS
- DISPLAY — SEE FIGURE 6.1.-A
- EXPECTED VALUES
- EST. NO. 1 -5 -30 -10 / NO. 2 -8
- ESTIMATE NO.
- DISPLAY RESULTS (Figure 6.1.B)

Return

1. The current set of probabilities (PROBS) in the user's model are displayed so that new estimates may be selected which differ from the current.

2. In interactive mode, the user is asked to specify new estimates for the probabilities of event outcomes in the current model. If no new estimates are given (a blank input line is entered by the user), then results computed from the current probabilities and previous estimates are displayed.

3. A new set of probabilities is normalized to sum to 100. The result of this calculation is displayed below the user's original input.

4. The user is allowed to make corrections before the results are computed.

5. A general routine is called to compute the expected values for options with the new set of probabilities (see chart 8.3)

6. The results of step 5 above are added to a table for estimated expected values. The number of estimates is incremented by one and the process (steps 2–6) is repeated.

MANUALLY CHANGE PROBS – EVACUATION

	NONE	P-300	P-2K	NP-2K	NP-6K
CURRENT PROBS (TRIAL 0)	53	6	14	17	1
PROBS, TRIAL 1	1	1	1	1	1
NORMALIZED	20	20	20	20	20
PROBS, TRIAL 2	10	1	5	6	5
NORMALIZED	37	4	19	22	19
PROBS, TRIAL 3	20	1	5	6	5
NORMALIZED	54	3	14	16	14
PROBS, TRIAL 4					
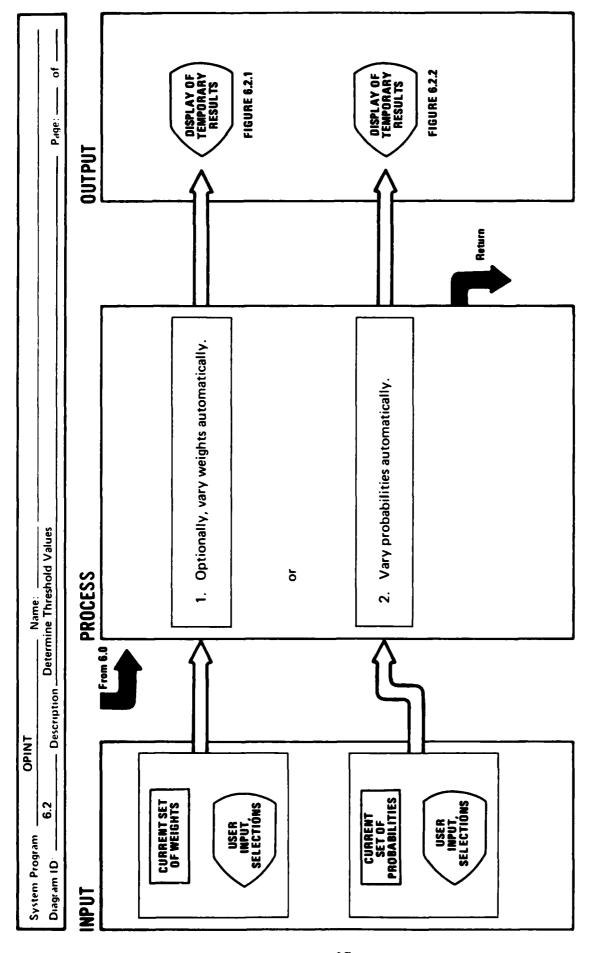
Figure 6.1-A

SENSITIVITY ANALYSIS:
EXAMPLE OF USER-INTERACTIVE DISPLAY OUTPUT
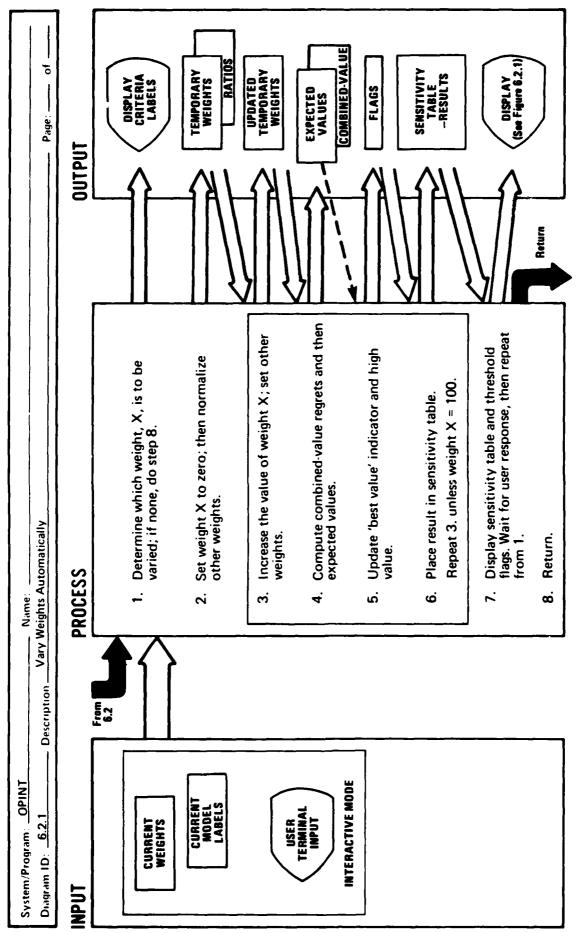
NOTE: Underlined values indicate user terminal input

MANUALLY CHANGE PROBS – EVACUATION

TRIAL	NORMAL	LOW PROF	MED PROF	EVAC PST
0	−27	−29	−26	−28
1	−39	−35	−26	−27
2	−36	−36	−26	−28
3	−27	−28	−26	−28

Figure 6.1-B

SENSITIVITY ANALYSIS: EXAMPLE OF OUTPUT FROM FUNCTION 6.1

INPUT

CURRENT SET OF WEIGHTS

USER INPUT, SELECTIONS

CURRENT SET OF PROBABILITIES

USER INPUT, SELECTIONS

PROCESS

From 6.0

1. Optionally, vary weights automatically.

or

2. Vary probabilities automatically.

Return

OUTPUT

DISPLAY OF TEMPORARY RESULTS

FIGURE 6.2.1

DISPLAY OF TEMPORARY RESULTS

FIGURE 6.2.2

The results of these operations are derived for display purposes only and are temporary values. The weights (WEIGHTS) and probabilities (PROBS) of the current model are never modified permanently by these operations.

INPUT

- CURRENT WEIGHTS
- CURRENT MODEL LABELS
- USER TERMINAL INPUT

INTERACTIVE MODE

From 6.2

PROCESS

1. Determine which weight, X, is to be varied; if none, do step 8.

2. Set weight X to zero; then normalize other weights.

3. Increase the value of weight X; set other weights.

4. Compute combined-value regrets and then expected values.

5. Update 'best value' indicator and high value.

6. Place result in sensitivity table. Repeat 3. unless weight X = 100.

7. Display sensitivity table and threshold flags. Wait for user response, then repeat from 1.

8. Return.

OUTPUT

- DISPLAY CRITERIA LABELS
- TEMPORARY WEIGHTS
  - RATIOS
- UPDATED TEMPORARY WEIGHTS
- EXPECTED VALUES
  - COMBINED-VALUE
- FLAGS
- SENSITIVITY TABLE –RESULTS
- DISPLAY (See Figure 6.2.1)

Return

The above procedure is an iterative one.

1. The user specifies which criterion weight he wants varied automatically. If no criterion name is entered, control is returned to the caller.

2. The weight for the specified criterion is set to sum to 100. This process preserves the current ratio of values (RATIOS) for the non-specified weights. (NORMALIZE, Chart 8.7)

3. The step increment for automatic variation is usually set at 10. The ratio values for the other weights is multiplied by the difference between 100 and WEIGHT X value.

4. Subroutines are called to calculate new combined regrets and expected values. (See Charts 8.4 and 8.3)

5. An indicator value is then set for each step increment which specifies the option with the lowest (best) expected value. A threshold flag is set whenever the actual value is better than previous ones.

6. The results are temporarily held in a table until displayed in 7.

7. Threshold arrows are displayed below the new best-value level increments.

38

SENSITIVITY ANALYSIS

EXPECTED VALUE WHEN WEIGHT OF EXPOSURE RISK IS:

	0	10	20	30	40	50	60	70	80	90	100
NORMAL	−22*	−23*	−25*	−26*	−27	−29	−30	−31	−33	−34	−35
LOWPROF	−32	−31	−31	−30	−29	−29	−28	−27	−27	−26	−25
MEDPROF	−44	−39	−35	−31	−27*	−23*	−19*	−15*	−11	−7	−3
EVACPST	−49	−44	−40	−35	−30	−25	−20	−15	−10*	−5*	−1*

Figure 6.2.1

SENSITIVITY ANALYSIS: EXAMPLE OF OUTPUT FROM FUNCTION 6.2.1

System Program: __OPINT__ Name: _____

Diagram ID __6.2.2__ Description __Vary Probabilities Automatically__ Page: ___ of ___

INPUT

CURRENT SET OF PROBABILITIES

LABELS FOR OUTCOMES, OPTIONS

TERMINAL INPUT

USER INTERACTIVE MODE

From 6.2

PROCESS

1. Determine which probability, X, is to be varied. If none, do 8.

2. Set prob. X to zero; then normalize other probs.

3. Increase the value of prob. X; set other probs.

4. Compute expected values.

5. Update 'best value' indicator and high value.

6. Place results in the sensitivity table. Repeat 3. unless prob. X = 100.

7. Display table and threshold flags. Wait for user response; repeat from 1.

8. Return.

OUTPUT

DISPLAY OUTCOME LABELS

TEMPORARY PROBS., RATIOS

TEMPORARY PROBS UPDATED

EXPECTED VALUE

FLAGS

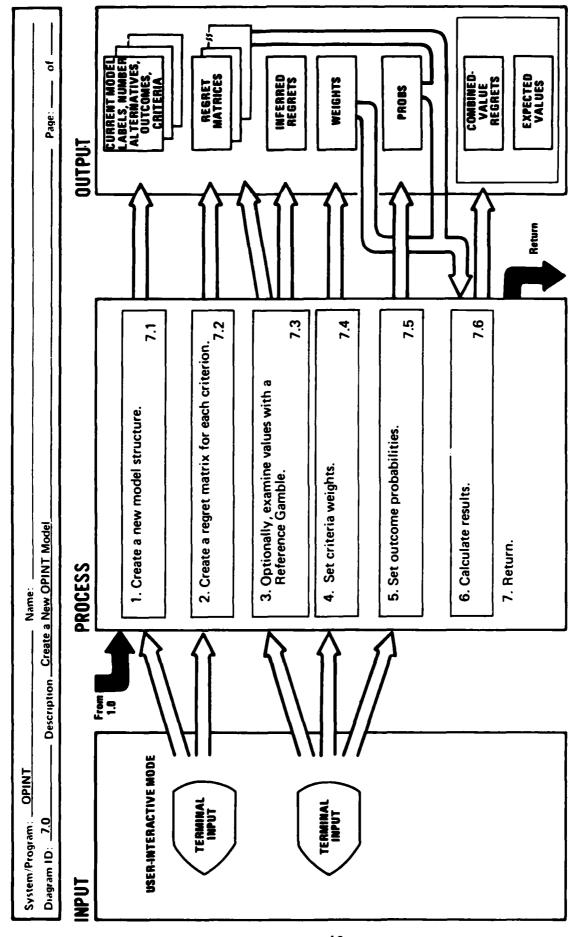**SENSITIVITY TABLE OF RESULTS**

DISPLAY (See Figure 6.2.2.)

Return

This procedure is an iterative one.

1. The user specifies which event outcome probability is to be varied. If no event name is specified, then processing control is returned to the calling routine.

2. The probability for the specified outcome is set to zero and the set of (PROBS) current probabilities is normalized. The values for the non-specified outcome probabilities are maintained ratios. (RATIOS)

3. A step increment value of 10 is used to automatically modify the value of Prob. X for sensitivity testing. Other probabilities are maintained in constant ratio such that all probabilities sum to 100.

4. Expected values for temporary probabilities are computed via subroutine. (See Chart 8.3)

5. The option considered to have the best expected value is indicated via a flag setting for each step increment. Another flag/indicator is set when a value is higher than the previous PAYOFF "best" value indicators. An arrow is shown later in the display to show this threshold value.
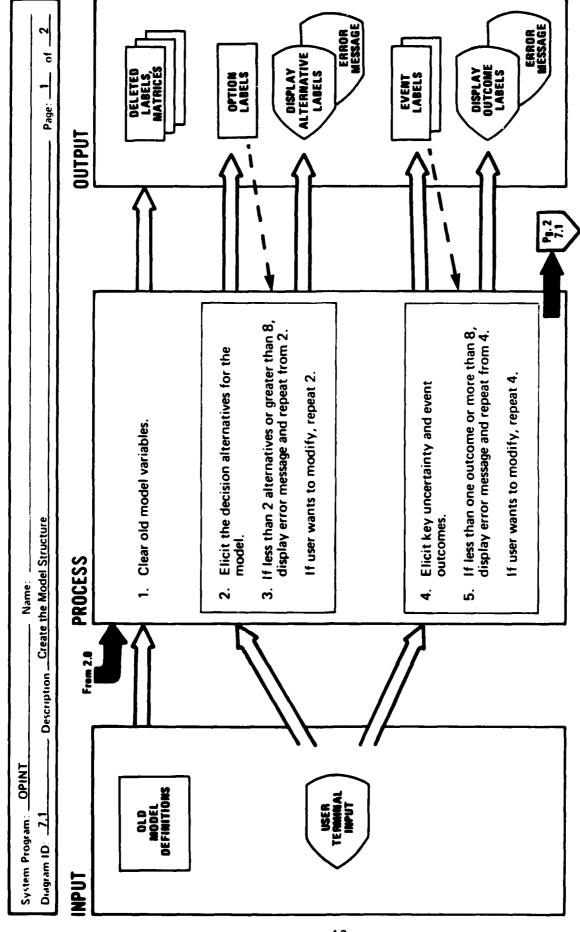
```
                        SENSITIVITY ANALYSIS


                EXPECTED VALUE WHEN PROBABILITY OF NONE IS:

             0      10     20     30     40     50     60     70     80     90     100

NORMAL     –57    –51    –45    –40    –34    –28    –23*   –17*   –11*    –6*     0*

LOWPROF    –50    –46    –42    –38    –34    –29    –25    –21    –17    –13     –9

MEDPROF    –27    –27    –27    –26*   –26*   –25*   –25    –25    –24    –24    –24

EVACPST    –26*   –26*   –26*   –27    –27    –28    –28    –28    –29    –29    –29

                                    ↑                      ↑
```

Figure 6.2.2

SENSITIVITY ANALYSIS: EXAMPLE OF OUTPUT FROM FUNCTION 6.2.2

INPUT

USER-INTERACTIVE MODE

TERMINAL INPUT

TERMINAL INPUT

From 1.0

PROCESS

1. Create a new model structure.	7.1
2. Create a regret matrix for each criterion.	7.2
3. Optionally, examine values with a Reference Gamble.	7.3
4. Set criteria weights.	7.4
5. Set outcome probabilities.	7.5
6. Calculate results.	7.6
7. Return.	

OUTPUT

CURRENT MODEL LABELS, NUMBER ALTERNATIVES, OUTCOMES, CRITERIA

REGRET MATRICES

INFERRED REGRETS

WEIGHTS

PROBS

COMBINED-VALUE REGRETS

EXPECTED VALUES

Return

This function causes the operating model definitions to be replaced by a new set of model definitions.

Charts 7.1 – 7.6 describe the subprocesses that are denoted here in steps 1-6. The subprocesses are executed in the order shown in the diagram.

PROCESS

1. Clear old model variables.

2. Elicit the decision alternatives for the model.

3. If less than 2 alternatives or greater than 8, display error message and repeat from 2.

   If user wants to modify, repeat 2.

4. Elicit key uncertainty and event outcomes.

5. If less than one outcome or more than 8, display error message and repeat from 4.

   If user wants to modify, repeat 4.

OUTPUT

DELETED LABELS, MATRICES

OPTION LABELS

DISPLAY ALTERNATIVE LABELS

ERROR MESSAGE

EVENT LABELS

DISPLAY OUTCOME LABELS

ERROR MESSAGE

Pg. 2 7.1

INPUT

OLD MODEL DEFINITIONS
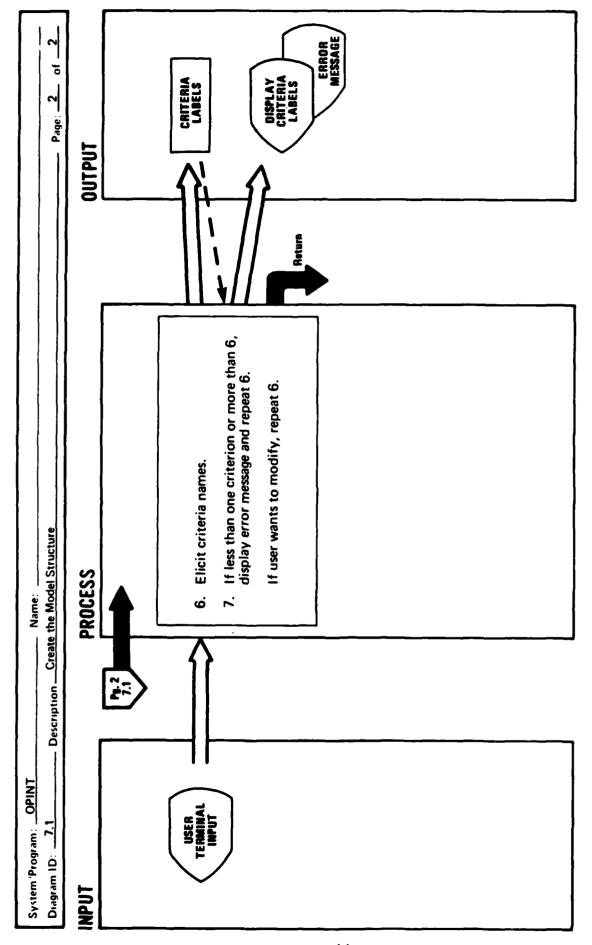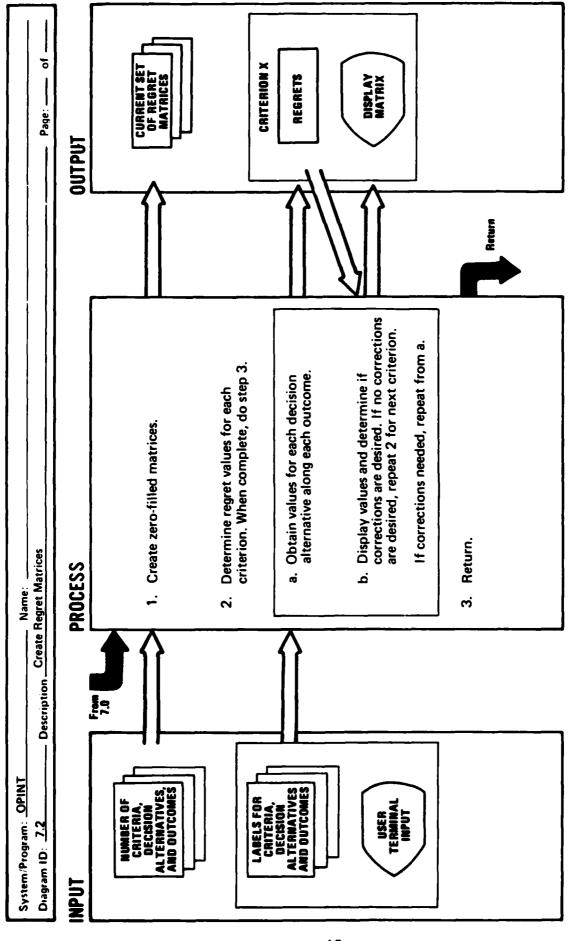
USER TERMINAL INPUT

From 2.0

1. Set variables to null value.

2. In interactive mode the user is requested to input decision alternative labels. A blank entry terminates the prompting and thereby, the number of decision alternatives is determined.

3. The number of decision alternatives is checked for validity. If not valid, the appropriate message is displayed and the elicitation procedure is repeated. If valid, the labels are displayed as a group and the user is given a change to modify these labels. Otherwise, processing continues at the next step.

Steps 4 and 5, 6 and 7 are handled in the same manner as steps 2 and 3.

43

OUTPUT

CRITERIA LABELS

DISPLAY CRITERIA LABELS

ERROR MESSAGE

PROCESS

Pg.2
7.1

6. Elicit criteria names.

7. If less than one criterion or more than 6, display error message and repeat 6.

   If user wants to modify, repeat 6.

Return

INPUT

USER TERMINAL INPUT

44

INPUT

NUMBER OF CRITERIA, DECISION ALTERNATIVES, AND OUTCOMES

LABELS FOR CRITERIA, DECISION ALTERNATIVES AND OUTCOMES

USER TERMINAL INPUT

PROCESS

From 7.0

1. Create zero-filled matrices.

2. Determine regret values for each criterion. When complete, do step 3.

   a. Obtain values for each decision alternative along each outcome.

   b. Display values and determine if corrections are desired. If no corrections are desired, repeat 2 for next criterion.

      If corrections needed, repeat from a.

3. Return.

OUTPUT

CURRENT SET OF REGRET MATRICES

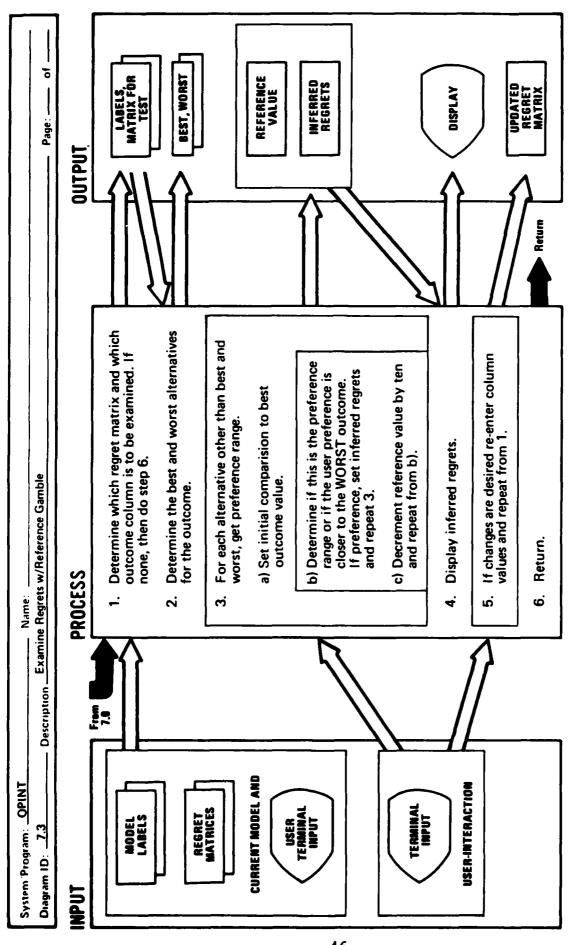**CRITERION X**

REGRETS

DISPLAY MATRIX

Return

1. Zero-filled arrays of the correct size at this step will allow the user to input values on the same display line with existing values, thus facilitating the entry process.
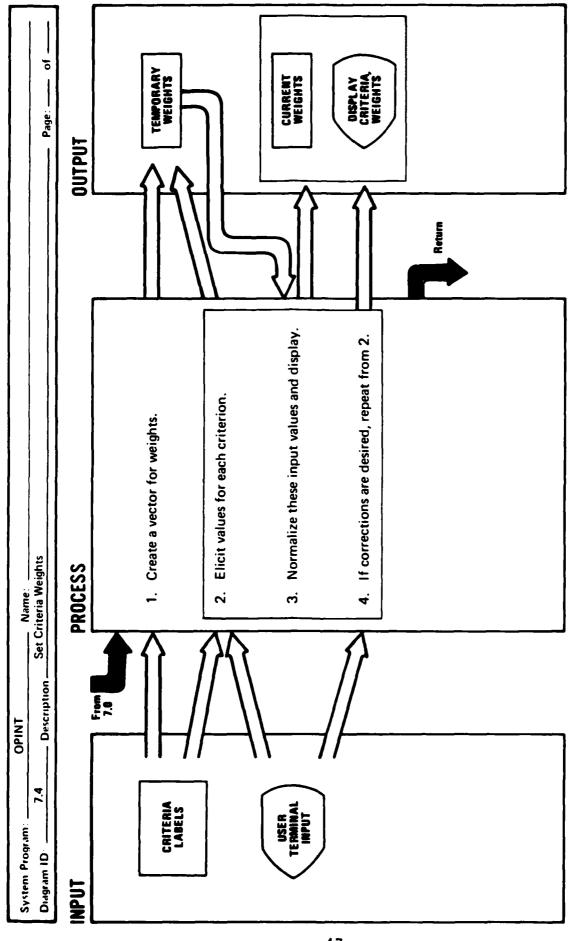
2. Regrets are entered for each criterion, one row (decison alternative) at a time.

Corrections, when needed, are also entered by rows.

45

INPUT

CURRENT MODEL AND
- MODEL LABELS
- REGRET MATRICES
- USER TERMINAL INPUT

USER-INTERACTION
- TERMINAL INPUT

From 7.0

PROCESS

1. Determine which regret matrix and which outcome column is to be examined. If none, then do step 6.

2. Determine the best and worst alternatives for the outcome.

3. For each alternative other than best and worst, get preference range.

   a) Set initial comparision to best outcome value.

   b) Determine if this is the preference range or if the user preference is closer to the WORST outcome. If preference, set inferred regrets and repeat 3.

   c) Decrement reference value by ten and repeat from b).

4. Display inferred regrets.

5. If changes are desired re-enter column values and repeat from 1.

6. Return.

OUTPUT

- LABELS, MATRIX FOR TEST
- BEST, WORST
- REFERENCE VALUE
- INFERRED REGRETS
- DISPLAY
- UPDATED REGRET MATRIX

Return

---

1. The user is prompted to input the name of the regret matrix (criterion label) and the outcome label for the column to be analyzed.

2. The best or minimum regret value in the selected column is determined for comparison tests. Next, the worst or maximum regret value is computed.

3. For each decision alternative regret value other than the best or worst, a probability p is selected (the preference value), by the user in interactive mode, which determines the user preference for that alternative-outcome event versus the probability p of the best alternative-outcome occurrence and 1−p probability of the worst alternative-outcome occurrence.
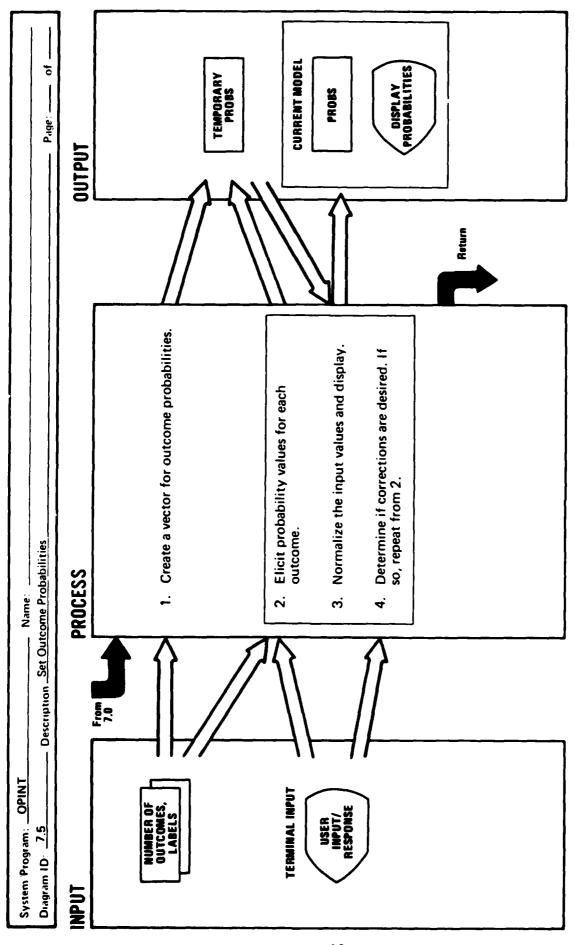
The probability p starts at 100 (or absolute 1.0) and is automatically decremented by 10 points each time the user is asked to indicate his preference value. If p is not the preference value, the user returns a blank response and another decrement of p is made. This process is repeated until p = 0 or the user indicates the specified p is preferred. At this point the inferred regret is 100−p times the worst regret value plus p times the best.

46

INPUT

CRITERIA LABELS

USER TERMINAL INPUT

PROCESS

From 7.0

1. Create a vector for weights.

2. Elicit values for each criterion.

3. Normalize these input values and display.

4. If corrections are desired, repeat from 2.

OUTPUT

TEMPORARY WEIGHTS

CURRENT WEIGHTS

DISPLAY CRITERIA WEIGHTS

Return

1. Create a vector of weights containing an element = 1 for each criterion.

2. Prompt the user for integer input values – a value for each criterion matrix. A conversion routine can check character to numeric translation to ensure valid input. (NUMS – Chart 8.6)
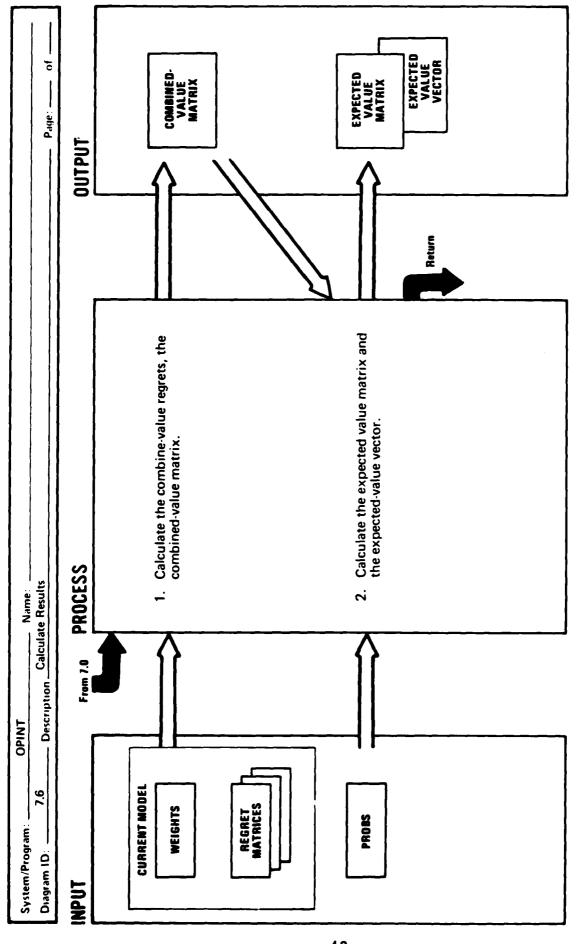
3. Take the relative weights input from the keyboard and normalize these so that the sum of the weights is 100. (NORMALIZE Chart 8.7)

INPUT

NUMBER OF OUTCOMES, LABELS

TERMINAL INPUT

USER INPUT/ RESPONSE

PROCESS

From 7.0

1. Create a vector for outcome probabilities.

2. Elicit probability values for each outcome.

3. Normalize the input values and display.

4. Determine if corrections are desired. If so, repeat from 2.

Return

OUTPUT

TEMPORARY PROBS

CURRENT MODEL

PROBS

DISPLAY PROBABILITIES

1. Create a vector of probabilities so that an element (=1) exists in the vector for each event outcome.

2. Prompt the user for numerical input values – integers between 0 and 100. Check for validity of input and conversion to numerical data. (NUMS, Chart 8.6)
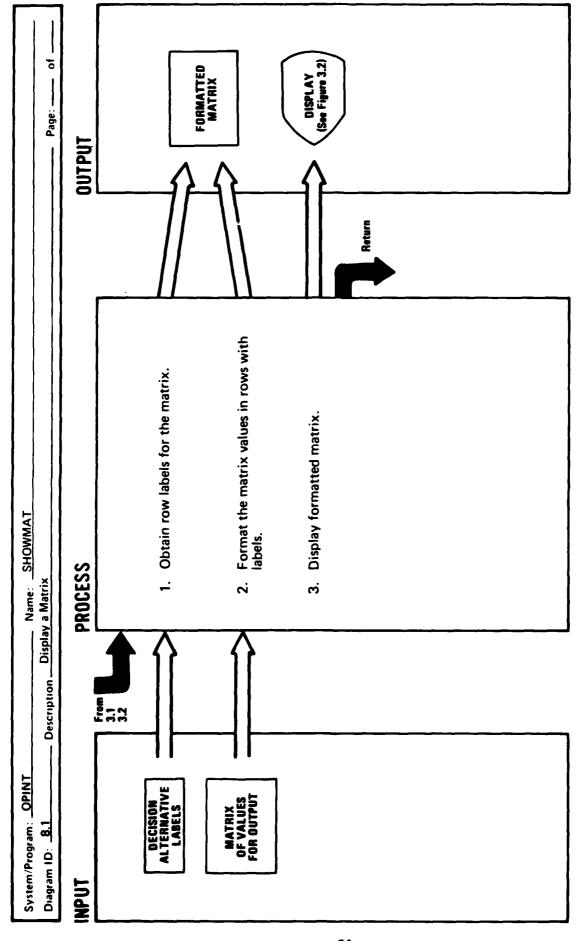
3. Invoke the normalization routine to ensure that the probabilities sum to 100 and have the same relative value as the keyboard input values. (NORMALIZE,

48

INPUT

CURRENT MODEL

- WEIGHTS
- REGRET MATRICES
- PROBS

PROCESS

From 7.0

1. Calculate the combine-value regrets, the combined-value matrix.

2. Calculate the expected value matrix and the expected-value vector.

Return

OUTPUT

- COMBINED-VALUE MATRIX
- EXPECTED VALUE MATRIX
- EXPECTED VALUE VECTOR

1. The combined-value matrix is formed by multiplying the weight for each criterion times the regret matrix for the criterion and then summing the position elements over each criterion.
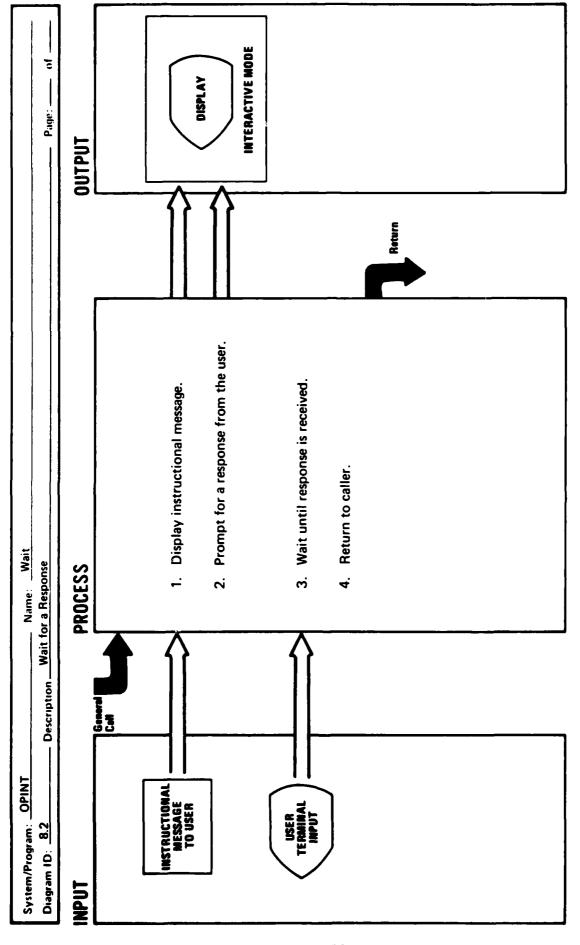
A single array is formed which is a combination of all the regrets weighted (Ref. CALCCV, Chart 8.4)

2. The weighted expected-value matrix is formed by multiplying each column of the combined-value regret matrix by the appropriate probability value for that outcome column.

The expected value vector is simply the row totals of this resulting matrix (Ref. CALCEV, Chart 8.3)

49

The following charts 8.1 – 8.7 describe generalized service routines for functions that are within the hierarchical structure of OPINT.

INPUT

DECISION
ALTERNATIVE
LABELS

MATRIX
OF VALUES
FOR OUTPUT

From
3.1
3.2

PROCESS

1. Obtain row labels for the matrix.

2. Format the matrix values in rows with labels.

3. Display formatted matrix.

Return

OUTPUT
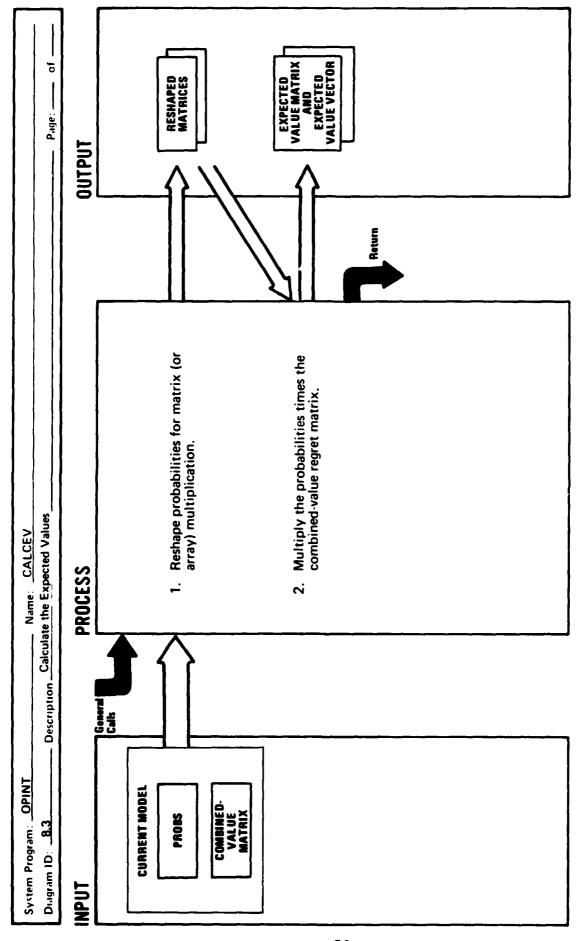
FORMATTED
MATRIX

DISPLAY
(See Figure 3.2)

1. Matrix row labels and matrix values (usually an array) are passed as input to this routine. Alternative names are the row labels usually.

2. Formatting the Matrix for display requires the concatenation of character and numerical values into character or display output.

51

INPUT

INSTRUCTIONAL MESSAGE TO USER

USER TERMINAL INPUT

General Call

PROCESS

1. Display instructional message.

2. Prompt for a response from the user.

3. Wait until response is received.

4. Return to caller.

Return
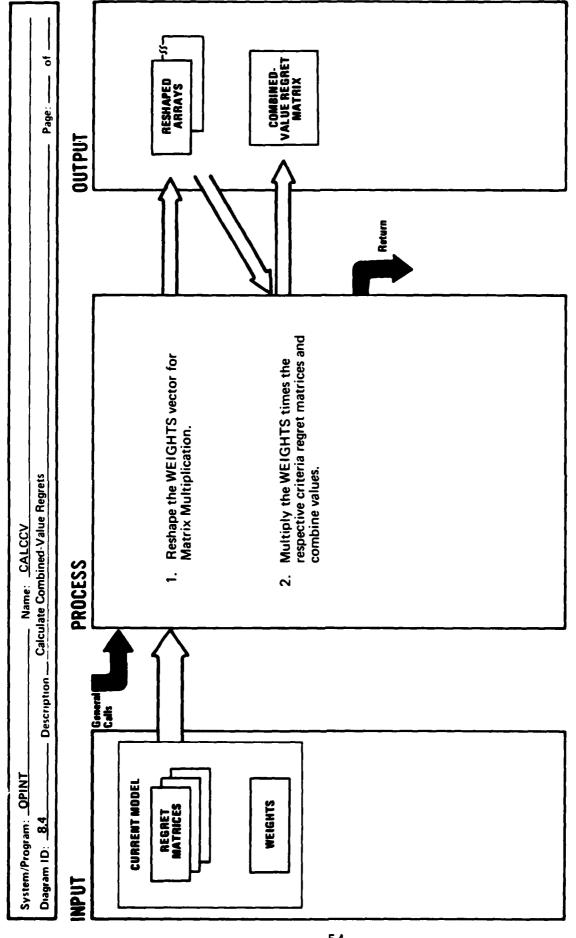
OUTPUT

DISPLAY

INTERACTIVE MODE

This routine is used for display purposes to enhance user control over changing displays.

3. Any keyboard input or a specific response may be requested of the user before processing can continue.

INPUT

CURRENT MODEL

PROBS

COMBINED-
VALUE
MATRIX

General
Calls

PROCESS

1. Reshape probabilities for matrix (or array) multiplication.

2. Multiply the probabilities times the combined-value regret matrix.

Return

OUTPUT

RESHAPED
MATRICES

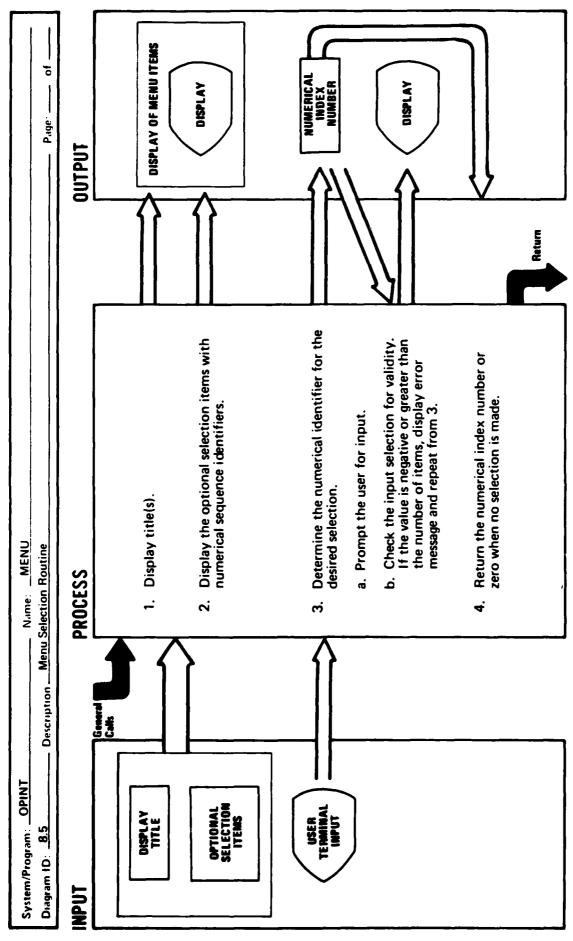EXPECTED
VALUE MATRIX
AND
EXPECTED
VALUE VECTOR

1. If matrix multiplication cannot occur in the system language by immediate instruction, then this step is omitted and step 2 is an iterative procedure — multiplying the appropriate probability times elements in the respective outcome column of the combined-value matrix.

INPUT

CURRENT MODEL

REGRET MATRICES

WEIGHTS

PROCESS

1. Reshape the WEIGHTS vector for Matrix Multiplication.

2. Multiply the WEIGHTS times the respective criteria regret matrices and combine values.

General Calls

Return

OUTPUT

RESHAPED ARRAYS

COMBINED-VALUE REGRET MATRIX

1. This step is required only for systems which allow single instruction matrix multiplication. Otherwise, step 2 is an iterative step.
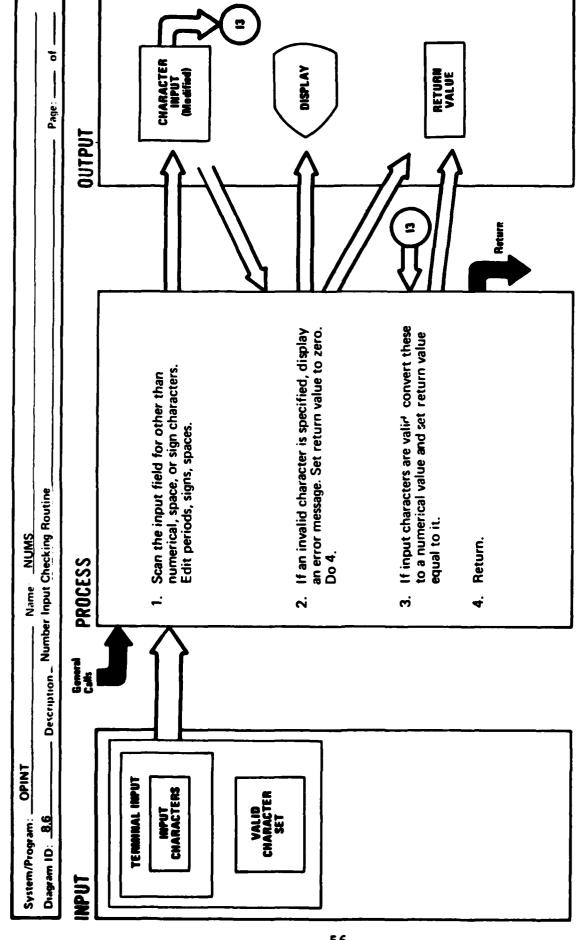
2. The elements in each criterion Matrix are multiplied by the respective criterion weight. Then the cell values are added so that the (i,j) – th elements of all criteria matrices are added together for a particular alternative i and event outcome j.

INPUT

- DISPLAY TITLE
- OPTIONAL SELECTION ITEMS
- USER TERMINAL INPUT

General Calls

PROCESS

1. Display title(s).

2. Display the optional selection items with numerical sequence identifiers.

3. Determine the numerical identifier for the desired selection.

   a. Prompt the user for input.

   b. Check the input selection for validity. If the value is negative or greater than the number of items, display error message and repeat from 3.

4. Return the numerical index number or zero when no selection is made.

OUTPUT

- DISPLAY OF MENU ITEMS
- DISPLAY
- NUMERICAL INDEX NUMBER
- DISPLAY

Return

3. Prompt the user for the item sequence number of the choice selection.

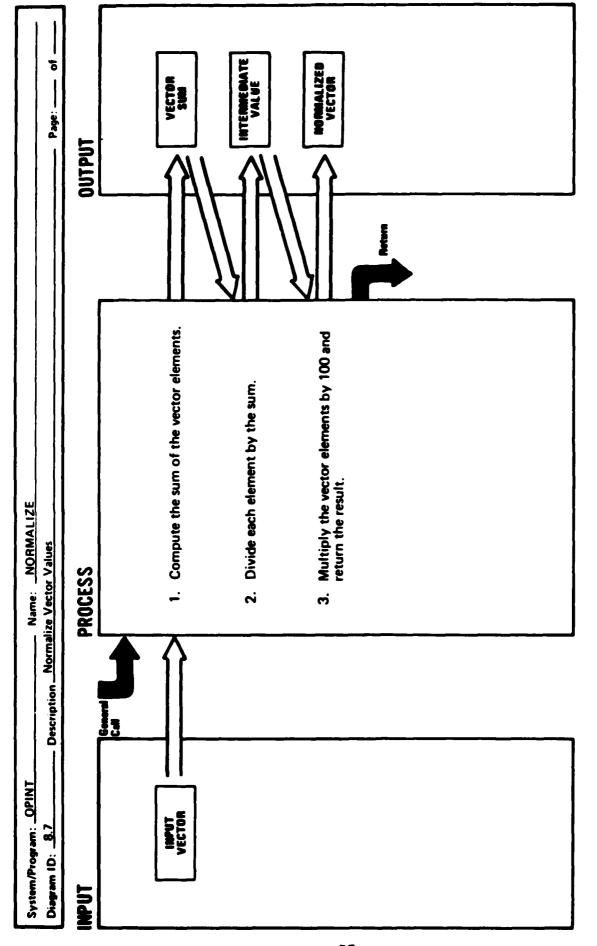Check the validity of the user input. (Ref. NUMS Chart 8.6)

1. The title is passed to this routine so that the display will remain in context with the processing function. For example, a title may be 'DISPLAY RESULTS . . . REGRET MATRICES'

2. The selections that describe what is optimal are passed as input and are displayed in a list or cookbook MENU format along with item sequence numbers.

55

INPUT

TERMINAL INPUT

INPUT CHARACTERS

VALID CHARACTER SET

General Calls

PROCESS

1. Scan the input field for other than numerical, space, or sign characters. Edit periods, signs, spaces.

2. If an invalid character is specified, display an error message. Set return value to zero. Do 4.

3. If input characters are valid, convert these to a numerical value and set return value equal to it.

4. Return.

OUTPUT

CHARACTER INPUT (Modified)

13

DISPLAY

RETURN VALUE

13

Return

This routine will not be required if system error checking routines interface with the standard keyboard-display input.

INPUT

INPUT
VECTOR

PROCESS

General
Call

1. Compute the sum of the vector elements.

2. Divide each element by the sum.

3. Multiply the vector elements by 100 and return the result.

Return

OUTPUT

VECTOR
SUM

INTERMEDIATE
VALUE

NORMALIZED
VECTOR

1. The input to this routine is a vector or row of numbers (otherwise, input will be converted to a numerical string of values). The sum of all the numbers in the vector is computed.

2. This step rescales all of the values to their percentage of the sum.

3. The normalized result from step 2 is typically multiplied by 100 to preserve percentage figures for output to the calling routines.